Implementasi Algoritma Pengenalan Wajah Untuk Mendeteksi *Visual Hacking*

Luisan William Alexander, Steven Ray Sentinuwo, Alwin Melkie Sambul
Teknik Informatika Universitas Sam Ratulangi Manado, Indonesia.
luisanalexander@gmail.com, steven@unsrat.ac.id, asambul@unsrat.ac.id

***Abstrak* – *Visual Hacking* merupakan sebuah isu keamanan dan privasi data yang perlu diperhatikan saat ini, dimana *visual hacking* berfokus pada pencurian informasi yang terpampang pada tampilan elektronik, seperti layar monitor komputer. Isu ini dapat terjadi ketika pengguna membiarkan informasi terpampang pada layar komputer sehingga dapat dilihat oleh siapa saja. Pada penelitian ini, dibuatlah sebuah aplikasi berbasis *computer vision* dengan tujuan mengimplementasikan algoritma pengenalan wajah *eigenface* untuk mendeteksi *visual hacking*. Penelitian ini menggunakan *framework* Viola-Jones dan fitur *Local Binary Pattern* (LBP) untuk mendeteksi wajah. Aplikasi yang dibuat dapat mengenali wajah pengguna dan akan mengeluarkan jendela peringatan jika terdeteksi wajah pengintip. Jika tidak terdeteksi wajah atau yang terdeteksi bukan wajah pengguna, aplikasi akan mengeluarkan jendela peringatan besar untuk menutupi informasi pada monitor. Aplikasi diuji pada ruangan dengan kondisi eksperimen terkontrol. Hasil pengujian menunjukkan bahwa aplikasi berhasil mendeteksi ancaman *visual hacking* dengan waktu kecepatan deteksi wajah 2.7003 detik serta tingkat akurasi pengenalan wajah 94%.**

Kata kunci : *Visual Hacking*, Pengenalan Wajah, *Computer Vision*, *Eigenface*, Viola-Jones, *Local Binary Pattern*

I. PENDAHULUAN

Teknologi membuat pekerjaan manusia menjadi lebih efisien, dan saat ini komputer telah menjadi bagian dari masyarakat perkotaan sebagai alat bantu pekerjaan, begitu pula hampir setiap perusahaan telah menerapkan dan menyediakan komputer untuk karyawannya sebagai alat bantu kerja demi mendapatkan hasil kerja yang optimal. Pastinya data-data penting perusahaan, data karyawan, data pelanggan, dan data lainnya disimpan pada komputer yang digunakan masing-masing personal. Harapan dari amannya data yang tersimpan di dalam teknologi merupakan suatu keharusan dari pengguna itu sendiri.

Visual hacking merupakan sebuah isu keamanan dan privasi data yang harus diperhatikan saat ini, karena berdasarkan penelitian dan eksperimen di berbagai negara berupa Tiongkok, Perancis, Jerman, India, Jepang, Korea Selatan, Inggris, dan Amerika oleh Dr. Larry Ponemon, Ph. D dan tim [1], dapat dilihat bahwa data rahasia personal dan perusahaan dapat terungkap melalui aktifitas *visual hacking*, dimana tingkat kesadaran akan aktifitas tersebut pada saat ini masih rendah. Rata-rata waktu yang diperlukan untuk melakukan aktifitas *visual hacking* hanya membutuhkan waktu kurang dari 15 menit, dengan tingkat kesuksesan mencapai 91%. Tingkat kesuksesan yang dimaksudkan dalam hal ini berhasil mendapatkan data dan informasi berupa data *login*, informasi finansial, dokumen istimewa dan rahasia, dan infromasi lainnya. Dr. Larry Ponemon, Ph. D dan tim [1] menyimpulkan bahwa 52% informasi yang sensitif dapat dilihat pada layar komputer karyawan sebuah perusahaan, dan 68% aktifitas *visual hacking* tidak diberhentikan atau disadari dari karyawan.

Visual hacking terjadi ketika karyawan membuat keputusan yang buruk dari bagaimana cara mereka mengakses data dan menampilkan informasi yang sensitif, yang kemudian dilihat dan dibaca oleh individu yang penasaran atau *hacker* yang berbahaya. Secara umum komputer rentan terkena *visual hacking* karena kurangnya kontrol dari pengguna dan komputer berada pada keadaan monitor yang sedang menyala, tidak berada dalam posisi *lock session login*, *brightness* monitor yang mati atau *blank window*. Posisi dimana komputer digunakan juga menentukan rentan atau tidaknya informasi dari komputer dapat dicuri melalui *visual hacking*.

Untuk itu perlu adanya kajian mengenai solusi untuk mendeteksi *visual hacking* yang lebih mendalam, dengan memanfaatkan algoritma tertentu dalam ilmu *computer vision* pada aplikasi, sebagai suatu solusi alternatif. Algoritma pengenalan wajah merupakan salah satu algoritma yang cocok untuk diterapkan.

II. LANDASAN TEORI

A. *Visual Hacking*

Visual hacking merupakan pengembangan dari *shoulder surfing* dimana *visual hacking* berfokus pada pencurian informasi yang terpampang pada barang elektronik, seperti layar monitor komputer/laptop atau gadget dan lainnya seperti pada Gambar 1.

Menurut Dr. Larry Ponemon dan tim [1][2][3], *Visual hacking* didefinisikan sebagai "*A low tech method of capturing sensitive, confidential and private information for unauthorized use*". Sesuai dengan penelitian dan eksperimen yang dilakukan oleh Dr. Larry Ponemon dan tim pada tahun 2015 (Amerika) dan tahun 2016 (Beberapa negara/global) di 46 perusahaan menghasilkan bahwa *visual hacking* hampir tidak disadari oleh para karyawan. Tingkat kesuksesan percobaan *visual hacking* mencapai 91%, 27% data dan informasi yang berhasil diretas mencakup *login credentials*, dokumen istimewa, dan

Gambar 1. Ilustrasi percobaan *visual hacking*

informasi finansial, 52% dari informasi yang sensitif berhasil diretas melalui pengelihatan secara visual di layar komputer karyawan dan *visual hacking* terjadi dengan sangat cepat kurang dari 15 menit dengan 68% percobaan *visual hacking* tidak diberhentikan oleh karyawan.

B. Viola-Jones Framework

Viola-Jones *framework* merupakan *framework* pendeteksian objek yang dipublikasikan pada tahun 2001 [4], algoritma dari *framework* pendeteksian Viola-Jones terdiri dari :

1. Membuat *integral image*
2. *Adaboost training*
3. *Cascade Classifiers*

Pendeteksian objek berdasasrkan Viola-Jones *framework* mengklasifikasikan citra berdasarkan fitur yang sederhana, tidak menggunakan piksel secara langsung untuk mengklasifikasikan citra. Alasan kuat menggunakan fitur sebagai pengklasifikasi karena fitur dapat mengkodekan atau membentuk domain *ad-hoc* yang sukar untuk dipelajari dengan menggunakan kuantitas *training* data yang terbatas, alasan kedua Viola dan Jones menggunakan fitur sebagai pengklasifikasi citra karena membuat proses komputasi lebih cepat dibandingkan dengan pengklasifikasian menggunakan piksel.

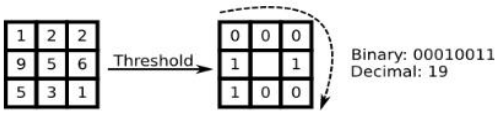
C. Local Binary Pattern

Local Binary Pattern (LBP) diperkenalkan oleh Ojala dkk [5] pada tahun 1996. Konsep dasar dari LBP yaitu menyimpulkan struktur lokal dari citra dengan membandingkan setiap piksel dengan piksel sekelilingnya. Cara perbandingan ini dilakukan dengan mengambil satu piksel tengah kemudian membandingkan nilainya dengan piksel sekelilingnya, jika nilai piksel yang mengelilingnya lebih besar atau sama dengan nilai pada piksel tengah maka piksel tersebut akan diberi nilai 1, sedangkan jika nilai dari piksel yang mengelilinginya kurang dari nilai piksel tengah maka diberi nilai 0, kemudian nilai dari setiap piksel pada citra akan menjadi angka biner.

Operator LBP yang diperkenalkan pertama kali bekerja dengan satu piksel tengah dengan delapan piksel yang mengelilinginya (lihat Gambar 2), nilai biner yang dihasilkan setelah perbandingan akan membentuk pola tekstur lokal. Setelah melakukan perbandingan nilai piksel keliling dengan nilai piksel tengah selajutnya dilakukan penyusunan delapan nilai biner searah jarum jam dan merubah nilai biner tersebut kedalam nilai desimal untuk menggantikan nilai piksel tengah.

D. Eigenface

M. Turk dkk [6] mendefinisikan *eigenfaces* sebagai hasil transformasi citra-citra wajah menjadi sebuah set fitur karakteristik wajah dalam bentuk *eigenvector* dari matirks kovarian citra-citra wajah tersebut. Dalam pendekatan dan pengenalan wajah dari [6], dijelaskan bahwa *eigenfaces* tidak melihat fitur-fitur tertentu pada wajah seperti mata, mulut, hidung, dan lain-lain untuk mendeteksi dan mengenali wajah, melainkan mengenal



Gambar 2. Operator LBP 3 x 3 *Pixel*

wajah dengan *eigenvector*, pendekatan pendeteksian dan pengenalan wajah dengan *eigenface* ini dibangun dengan tujuan agar proses pendeteksian dan pengenalan wajah menjadi cepat, simpel, dan akurat di dalam lingkungan yang terbatas seperti ruang kantor atau ruangan pada rumah. Konsep dari pendekatan dengan *eigenface* ini ialah dengan mengekstrak informasi relevan dari citra wajah dengan cara *encoding*, kemudian membandingkan hasil *encoding* citra wajah tersebut dengan citra wajah yang telah di-*encode* sebelumnya, dalam pendekatan ini informasi yang di-*encode* dari citra ialah *eigenvector* dari matriks kovarian citra-citra wajah.

M. Turk dkk [6] merangkum 5 langkah dari proses pengenalan wajah dengan *eigenface*, yaitu :

1. Inisialisasi : membutuhkan *dataset* dari citra-citra wajah yang nantinya akan di-*train* dan menghasilkan perhitungan *eigenfaces* yang akan mendefiniskan "*face-space*".
2. Ketika citra wajah baru dimasukkan, citra wajah tersebut dikalkulasi bobot vektornya dengan memproyeksikan citra tersebut ke dalam sub-ruang yang disebut dengan ("*face space*") yang dibentangi seluruh permukaannya oleh *eigenfaces*.
3. Tahap selanjutnya, menentukan apakah citra masukan tersebut adalah wajah dengan melakukan pengecekan yang melihat apakah citra tersebut cukup menyerupai dengan yang ada dalam "*face space*".
4. Jika itu adalah wajah, dilakukan pengklasfikasian citra wajah sebagai wajah yang dikenali atau tidak.
5. (Opsional) jika wajah yang tidak diketahui dideteksi berkali-kali, maka kalkulasi bobot karakteristik wajah tersebut kemudian hasil kalkulasi tersebut dihitung sebagai wajah yang dikenali.

Gambar 3 adalah citra *eigenface* pada penelitian [6].

E. Deteksi Wajah (Face Detection)

Face Detection merupakan teknologi komputer yang digunakan beberapa sistem dan atau aplikasi untuk mendeteksi wajah. Teknologi *face detection* dibangun dengan algoritma tertentu yang berfokus pada deteksi dari wajah manusia. Di dalam teknologi pengenalan wajah, *face detection* merupakan tahap awal pemrosesan untuk mengenali wajah seseorang, dimana *face detection* menentukan dimana bagian wajah yang muncul pada citra masukan. Keberhasilan dari *face detection* memiliki tingkat pengaruh yang tinggi dalam performa dan kegunaan dari suatu sistem pengenalan wajah [7].

Gambar 3. Citra *eigenface* pada penelitian M. Turk

F. Pengenalan Wajah

Face Recognition merupakan pengembangan dari teknologi *face detection* dimana teknologi ini dapat menghasilkan/*generate* wajah dari hasil tangkapan kamera dan melakukan deteksi persamaan wajah dengan data wajah yang diketahui komputer, sehingga komputer dapat mengenali dan atau mengetahui keberadaan seseorang. *Face Recognition* dikategorikan menjadi 3 kategori, yaitu *verification*, *identification*, dan *watch*, Mounica [8]

G. OpenCV

OpenCV (*Open Computer Vision Library*) [9] merupakan *library* perangkat lunak *open source* yang memiliki lisensi *BSD-licensed product*. OpenCV memiliki lebih dari 2500 algoritma yang telah di optimasi disediakan untuk menangani hal mengenai *computer vision* dan *machine learning*. Algoritma yang ada dapat digunakan untuk mendeteksi wajah, mengenali wajah, mengidentifikasi objek, dan lain-lain. Di dalam OpenCV, untuk algoritma pengenalan wajah yang disediakan sampai pada saat penelitian ditulis, telah tersedia tiga algoritma pengenalan wajah, diantaranya *Eigenface*, *Fisherface*, dan *Local Binary Pattern Histogram* (LBPH).

III. METODOLOGI PENELITIAN

A. Skenario yang Diharapkan

Pengguna menjalankan aplikasi pada komputer dengan komputer berada pada keadaan ruangan yang terkontrol dengan pencahayaan yang baik dan konsisten. Aplikasi berjalan pada komputer tanpa menggangu pekerjaan pengguna, ketika ada wajah selain pengguna yang tertangkap pada *webcam* komputer maka aplikasi akan memberikan peringatan kepada pengguna dengan menurunkan tingkat kecerahan monitor atau memberikan *pop up window* peringatan. Pengguna dapat menonaktifkan aplikasi yang sedang berjalan untuk menghindari peringatan ketika wajah selain pengguna berhasil dideteksi aplikasi. Ketika pengguna meninggalkan komputer aplikasi akan memberikan peringatan juga pada layar monitor dan menutup data yang terpampang pada monitor dengan *pop up window* atau menurunkan tingkat kecerahan layar monitor saja, setelah pengguna kembali semua peringatan hilang dan komputer siap digunakan kembali.

B. Metode Pengumpulan Data

Data dari penelitian ini berupa citra RGB yang diambil dari *webcam*, dimana pada citra tersebut terdapat objek wajah yang akan digunakan untuk proses pengenalan wajah. Citra yang diambil beserta objek wajah didalamnya berdasarkan dengan posisi tubuh nyaman sesorang (kondisi tegap dan jarak nyaman dari layar) yang sedang menggunakan komputer 14 inci pada ruangan dengan kondisi yang terkontrol pada tingkat pencahayaan 160lx.

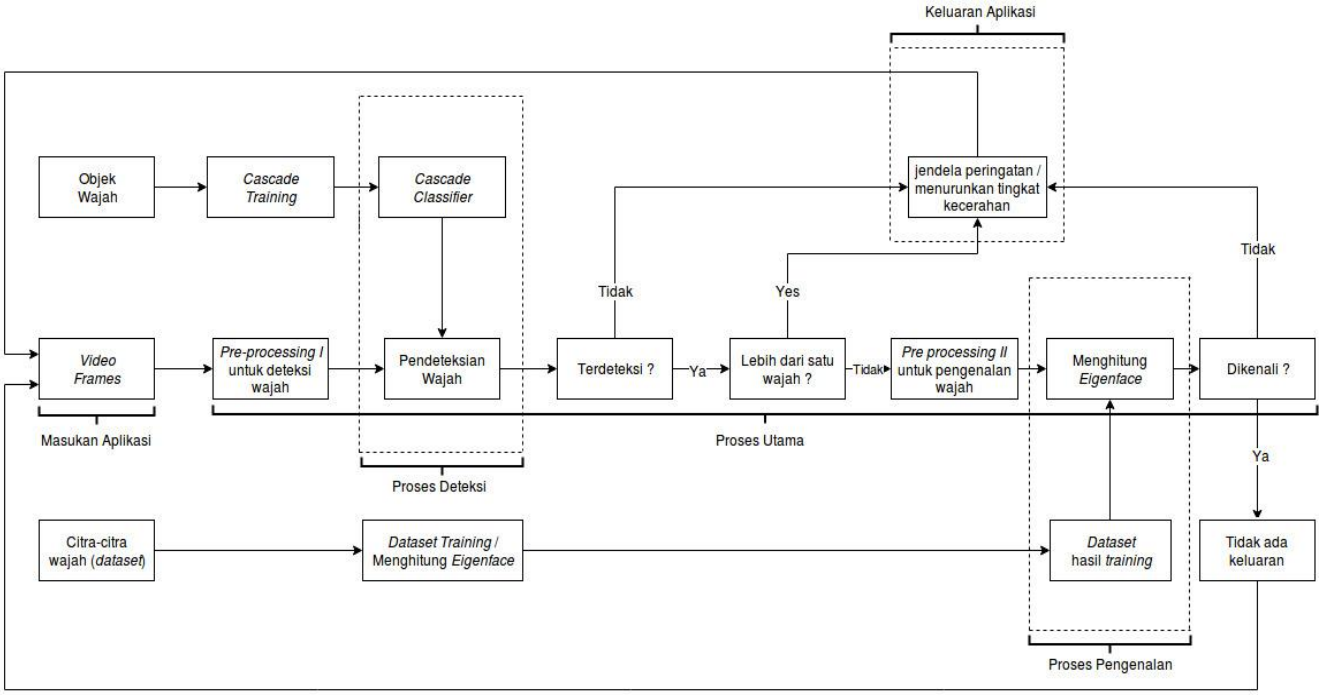
C. Perancangan Aplikasi

Aplikasi dibuat menggunakan library *OpenCV* dan ditulis dengan bahasa pemrograman C/C++. Gambar 4 merupakan blok diagram cara kerja aplikasi secara detail. Berdasarkan blok diagram pada Gambar 4, proses kerja aplikasi dibagi menjadi lima bagian besar, yaitu :

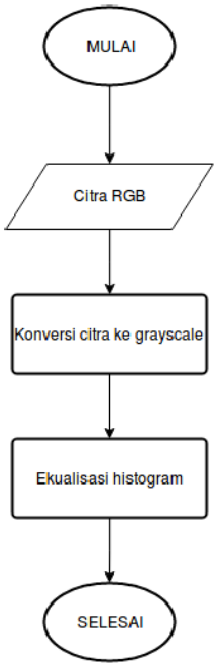
1. *Pre-processing I* untuk Deteksi
2. Deteksi Wajah
3. *Pre-processing II* untuk Pengenalan
4. Pengenalan Wajah
5. Logika Aplikasi

Tahap sebelum aplikasi dijalankan, terdapat dua keperluan penting untuk menyukseskan pengenalan wajah pada aplikasi, yaitu menyediakan *Cascade Classifier* yang sudah terdapat di OpenCV dan citra *dataset* yang sudah dilakukan *training* terlebih dahulu, algoritma *eigenface* digunakan dalam *training dataset*, total terdapat 48 citra yang berukuran 329 x 365 px yang dijadikan *dataset*.

Diawali dengan *Pre-processing I* citra masukan RGB dari *webcam* dikonversi ke *graysacle* terlebih dahulu dan dilakukan ekualisasi histogram (lihat Gambar 5).



Gambar 4. Blok Diagram Cara Kerja Aplikasi

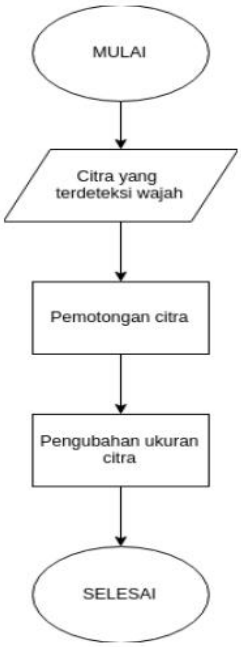


Gambar 5. *Flowchart Pre-processing I*

Selanjutnya, dilakukan proses pendeteksian wajah pada citra keluaran dari *pre-processing I*. Proses pendeteksian wajah dilakukan dengan metode *Viola-Jones* dengan fitur yang digunakan adalah fitur LBP.

Jika terdeteksi wajah pada citra, selanjutnya dilakukan tahap p*re-processing II* untuk pengenalan wajah. Tahap ini terdiri dari pemotongan area citra yang terdeteksi wajah, kemudian diubah ukurannya untuk disesuaikan dengan ukuran citra wajah pada *dataset*, Gambar 6 adalah *flowchart* tahap p*re-processing II*.

Setelah tahap *pre-processing II* selesai, dilakukanlah proses pengenalan wajah dengan menggunakan algoritma *eigenface*. Algoritma disediakan OpenCV dan dipanggil menggunakan method `createEigenfaceRecognizer()`

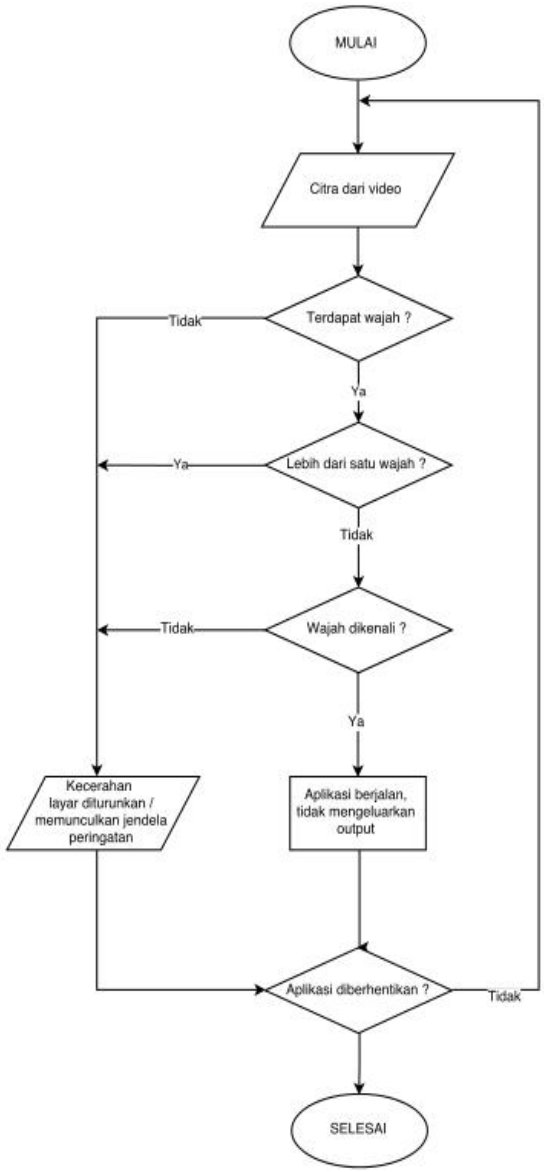


Gambar 6. *Flowchart Pre-processing II*

dengan fungsi `train()` untuk *training* citra dan `predict()` untuk melakukan prediksi pengenalan wajah. Di dalam proses ini citra masukan dari *pre-processing II* dilakukan perhitungan *eigenface* juga, sama seperti pada citra *dataset* yang telah di-*training*, tahap perhitungan yang pertama ialah citra diambil nilai tengahnya atau *mean*, kemudian dicari selisih antara *training image* dengan *mean*, lalu menghitung nilai matriks kovarian, dari matriks kovarian selanjutnya menghitung nilai *eigenvalue* dan *eigenvector* yang digunakan untuk menentukan nilai *eigenface*. Tahap terakhir, setelah nilai *eigenface* citra masukan dihitung, selanjutnya dilakukan identifikasi citra masukan dengan citra pada *dataset* dengan menggunakan metode *Ecluidean Distance.*

Hal penting yang terdapat dalam penelitian ini ialah logika aplikasi yang dibuat, dimana untuk mendukung teknologi pengenalan wajah mendeteksi *visual hacking* terdapat racikan logika di dalamnya. Gambar 7 merupakan *flowchart* dari logika aplikasi.

Masukan dari aplikasi merupakan citra video *stream*, jika pada citra masukan terdapat wajah yang tidak lebih dari satu dan dikenali, maka aplikasi tidak akan



Gambar 7. *Flowchart* Logika Aplikasi

mengeluarkan keluaran. Tetapi jika :

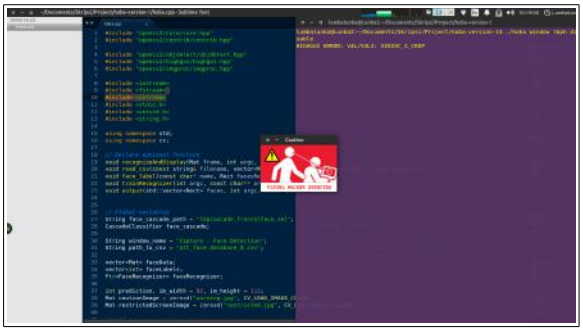
1. Aplikasi tidak mendeteksi adanya wajah.
2. Mendeteksi adanya wajah tetapi lebih dari satu dan tidak dikenali.
3. Mendeteksi satu wajah tetapi tidak mengenali wajah tersebut.

Maka secara otomatis aplikasi akan mengeluarkan keluaran berupa jendela peringatan atau menurunkan tingkat kecerahan layar monitor agar informasi yang ditampilkan di layar tidak mudah terbaca. Gambar 8 dan 9 merupakan keluaran jendela peringatan aplikasi.

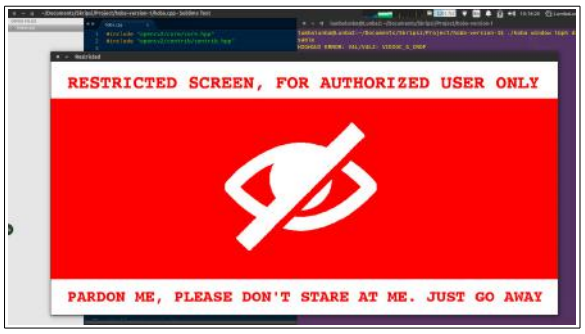
D. Desain Eksperimen Pengujian

Hasil impelementasi algoritma pengenalan wajah pada aplikasi, diuji pada desain eksperimen seperti berikut:

1. Ruang uji aplikasi memiliki latar belakang yang konsisten.
2. Ruang uji aplikasi memiliki tingkat pencahayaan 160lx yang konsisten.
3. Posisi deteksi ancaman *visual hacking* dari pengintip, adalah posisi pengintip yang berada pada sisi sebelah kanan pengguna.
4. Hasil pengujian deteksi ancaman *visual hacking* pengintip pada sisi kanan pengguna diasumsikan hasilnya sama dengan posisi pengintip ketika berada pada sisi sebelah kiri pengguna.
5. Posisi wajah bukan pengguna dan pengguna pada saat pengujian, bertatapan langsung dengan *webcam* sehingga wajah dapat dideteksi dan keadaan kepala berada dalam keadaan tegap.
6. Jarak pengguna terhadap komputer merupakan jarak posisi nyaman pengguna ketika menggunakan komputer.
7. Jarak pengintip yang diuji ialah jarak selama wajah pengintip dapat dideteksi, semua hasil uji pada variasi jarak yang dapat dideteksi diasumsikan sama hasilnya.



Gambar 8. Keluaran jika terdeteksi pemilik dan pengintip



Gambar 9. Jika hanya pengintip saja atau tidak ada wajah

IV. HASIL DAN PEMBAHASAN

Hasil implementasi algoritma pada aplikasi diuji berdasarakan desain eksperimen yang dibuat. Melalui desain eksperimen tersebut, yang akan diuji ialah kecepatan dan akurasi aplikasi dalam hal mendeteksi *visual hacking*. Pengujian melibatkan dua orang sebagai pengintip (bukan pengguna/BP-A/BP-B) dan pemilik komputer (P). Perangkat keras yang digunakan dalam pengujian memiliki spesifikasi sebagai berikut :

- Laptop/komputer dengan ukuran layar 14 inci dan resolusi 1366 x 768.
- Kamera *webcam built-in* dengan resolusi 1280 x 720 HD @ 30FPS.
- 4th *Generation* Intel Core i5 *processor* (*UProcessor Line*).
- Memori RAM tipe DDR3L dengan kapasitas 4GB.

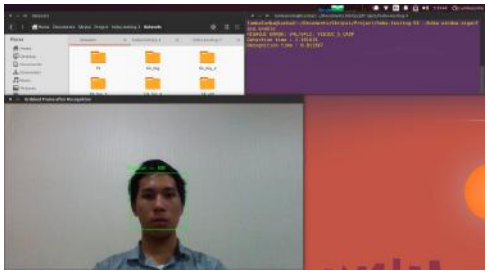
Sesuai dengan desain eksperimen pada BAB III dihasilkanlah skenario pengujian seperti pada Tabel 1.

Tabel 1. Daftar Skenario Uji

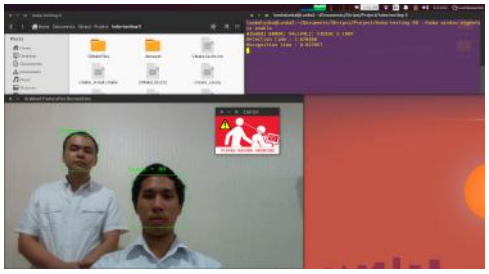
No.	Kondisi	Banyak Pengujian
1.	Keadaan komputer hanya ada pengguna	10x
2.	Keadaan komputer hanya ada pengintip (bukan pengguna) A	10x
3.	Keadaan komputer hanya ada pengintip (bukan pengguna) B	10x
4.	Terdapat pengintip (bukan pengguna) A dari sebelah pengguna	10x
5.	Terdapat pengintip (bukan pengguna) B dari sebelah pengguna	10x

A. Pengujian Aplikasi

Setiap skenario pengujian, aplikasi akan dieksekusi dan diakhiri sebanyak 10x untuk menghitung kecepatan dan akurasi aplikasi mendeteksi dan mengenali wajah pengguna (P) dan bukan pengguna (BP-A/BP-B), berdasarkan proses pendeteksian dan pengenalan dari masukan citra paling pertama pada saat aplikasi dijalankan. Gambar 10 dan Gambar 11 merupakan beberapa Gambar *screenshot* pengujian.



Gambar 10. Pengujian Deteksi Pemilik



Gambar 11. Pengujian Deteksi Pemilik dan BP-A

Tabel 2. Tabel Pengujian Kecepatan Waktu Deteksi

Skenario	Iterasi / Waktu (t1)										
	1	2	3	4	5	6	7	8	9	10	Rata-rata
P	2.7018 s	2.6608 s	2.7296 s	2.6766 s	2.7526 s	2.7120 s	2.7203 s	2.7265 s	2.6736 s	2.7510 s	**2.7104 s**
BP-A	2.6732 s	2.6873 s	2.6818 s	2.6657 s	2.7452 s	2.6745 s	2.7582 s	2.6525 s	2.6629 s	2.6976 s	**2.6895 s**
BP-B	2.6660 s	2.6766 s	2.6632 s	2.6618 s	2.6852 s	2.6603 s	2.6784 s	2.6677 s	2.6948 s	2.6625 s	**2.6761 s**
P & BP-A	2.6704 s	2.6617 s	2.6387 s	2.6471 s	2.6397 s	2.6551 s	2.6643 s	2.6627 s	2.6492 s	2.6515 s	**2.6540 s**
P & BP-B	2.6603 s	2.6925 s	2.6906 s	2.6756 s	2.6750 s	2.6815 s	2.6934 s	2.6897 s	2.6851 s	2.6864 s	**2.6830 s**
Rata-rata waktu deteksi											**2.6826 s**

Tabel 3. Tabel Pengujian Kecepatan Waktu Pengenalan

Skenario	Iterasi / Waktu (t2)										
	1	2	3	4	5	6	7	8	9	10	Rata-rata
P	0.0113 s	0.0094 s	0.0094 s	0.0095 s	0.0111 s	0.0093 s	0.0102 s	0.0099 s	0.0094 s	0.0131 s	**0.0102 s**
BP-A	0.0095 s	0.0094 s	0.0097 s	0.0094 s	0.0103 s	0.0093 s	0.0098 s	0.0094 s	0.0095 s	0.0093 s	**0.0095 s**
BP-B	0.0093 s	0.0096 s	0.0097 s	0.0098 s	0.0093 s	0.0093 s	0.0094 s	0.0093 s	0.0100 s	0.0094 s	**0.0095 s**
P & BP-A	0.0278 s	0.0372 s	0.0240 s	0.0271 s	0.0249 s	0.0368 s	0.0283 s	0.0324 s	0.0240 s	0.0303 s	**0.0292 s**
P & BP-B	0.0294 s	0.0239 s	0.0274 s	0.0262 s	0.0249 s	0.0230 s	0.0237 s	0.0235 s	0.0244 s	0.0258 s	**0.0252 s**
Rata-rata waktu pengenalan											**0.0167 s**

Keseluruhan 5 skenario telah diuji berdasarkan desain eksperimen terhadap aplikasi. Perhitungan waktu rata-rata kecepatan deteksi dan pengenalan beserta akurasinya didasarkan pada proses deteksi dan pengenalan pada citra yang paling pertama masuk dari *webcam* ketika aplikasi dijalankan yang proses tersebut dilkakuan sebanyak 10 iterasi. Tabel 2 dan 3 adalah tabel perhitungan waktu rata-rata kecepatan deteksi wajah dan pengenalan wajah.

Waktu rata-rata kecepatan deteksi wajah yang didapatkan ialah **2.6826 detik** dan waktu rata-rata kecepatan pengenalan wajah yang didapatkan ialah **0.0167 detik**. Selanjutnya waktu pada setiap iterasi dan skenario yang sama pada tabel pengujian, kecepatan waktu deteksi dan waktu pengenalan diakumulasi dan kemudian dihitung rata-ratanya untuk mendapatkan waktu rata-rata kecepatan deteksi *visual hacking*.

Hasil perhitungan menunjukkan waktu kecepatan aplikasi untuk mendeteksi *visual hacking* pada pengujian berdasarkan desain eksperimen dan perangkat keras yang digunakan, didapatkan waktu rata-rata sebanyak **2.7003 detik.** Pada saat yang bersamaan pada pengujian kecepatan waktu pengenalan wajah, dicatat pula akurasi pengenalan wajah apakah benar atau salah.

Tabel 4 merupakan tabel perhitungan waktu rata-rata kecepatan deteksi *visual* hacking.

Berdasarkan perhitungan, didapatkan presentasi akurasi pengenalan wajah aplikasi untuk mendeteksi *visual hacking* adalah **94%**. Tabel 5 merupakan tabel pengujian keberhasilan pengenalan wajah (akurasi).

B. Analisa Hasil Implementasi Algoritma

Pada proses pendeteksian wajah, dapat terjadi kegagalan deteksi dikarenakan beberapa faktor. Berikut adalah faktor-faktor yang mempengaruhi kegagalan deteksi :

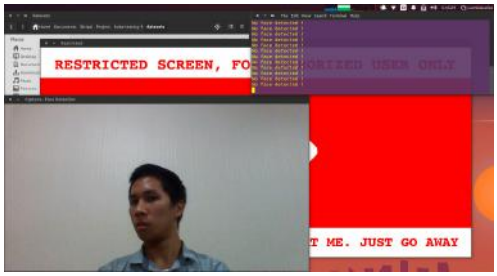
1. Putar Kepala (lihat Gambar 12).
2. Kemiringan Kepala (lihat Gambar 13).
3. Posisi kepala bagaimana citra diambil, pada Gambar 14 dapat dilihat bahwa, webcam hanya menangkap setengah bagian wajah saja (lihat Gambar 14).
4. Keadaan ruangan yang tidak terkontrol, dapat dilihat pada gambar 15, lampu pada latar belakang mempengaruhi citra area wajah yang dikenali algoritma pendeteksian (lihat Gambar 15).

Tabel 4. Tabel Uji Kecepatan Deteksi *Visual Hacking*

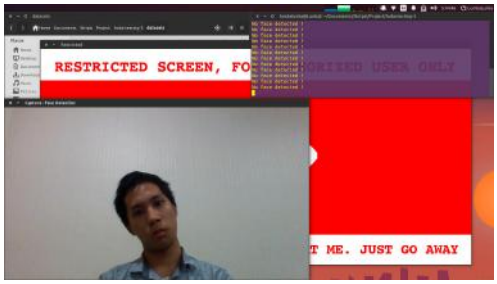
Skenario	Iterasi / Waktu (t)										
	1	2	3	4	5	6	7	8	9	10	Rata-rata
P	2.7131 s	2.6702 s	2.7390 s	2.6861 s	2.7637 s	2.7213 s	2.7305 s	2.7364	2.6830 s	2.7641 s	**2.7207 s**
BP-A	2.6827 s	2.6967 s	2.6915 s	2.6751 s	2.7555 s	2.6838 s	2.7683 s	2.6619 s	2.6724 s	2.7069 s	**2.6994 s**
BP-B	2.6753 s	2.7756 s	2.6729 s	2.6716 s	2.6945 s	2.6696 s	2.6878 s	2.6770 s	2.7048 s	2.6719 s	**2.6901 s**
P & BP-A	2.6978 s	2.6989 s	2.6627 s	2.6742 s	2.6630 s	2.6919 s	2.6923 s	2.6951 s	2.6732 s	2.6815 s	**2.6834 s**
P & BP-B	2.6897 s	2.7164 s	2.7180 s	2.7018 s	2.6999 s	2.7045 s	2.7171 s	2,7132 s	2.7095 s	2.7122 s	**2.7082 s**
Rata-rata waktu deteksi *visual hacking*											**2.7003 s**

Tabel 5. Tabel Uji Keberhasilan Pengenalan (akurasi)

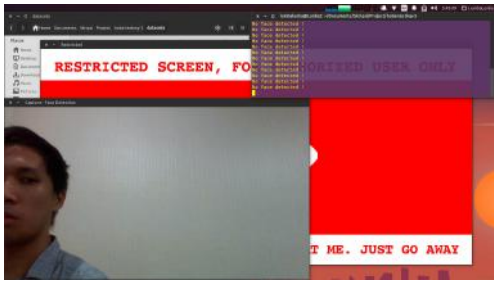
Skenario	Iterasi / Pengenalan										
	1	2	3	4	5	6	7	8	9	10	Akurasi
P	Benar	Benar	Benar	Benar	Benar	Benar	Benar	Benar	Benar	Benar	**100%**
BP-A	Benar	Benar	Benar	Benar	Benar	Benar	Benar	Benar	Benar	Benar	**100%**
BP-B	Benar	Benar	Benar	Benar	Benar	Benar	Benar	Benar	Benar	Benar	**100%**
P & BP-A	Benar	Benar	Benar	Benar	Salah	Salah	Benar	Benar	Salah	Benar	**70%**
P & BP-B	Benar	Benar	Benar	Benar	Benar	Benar	Benar	Benar	Benar	Benar	**100%**
Presentase akurasi pengenalan wajah											**94%**



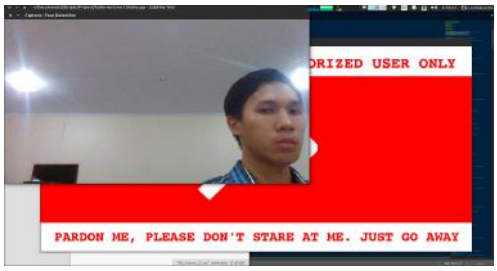
Gambar 12. Tidak Terdeteksi Wajah Karena Putar Kepala



Gambar 13. Gagal Deteksi Karena Kemiringan Kepala

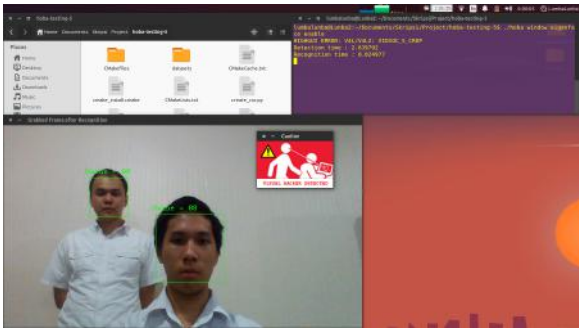


Gambar 14. Gagal Deteksi Karena Posisi Kepala



Gambar 15. Kondisi Ruangan Tidak Terkontrol

Dalam beberapa kondisi keadaan wajah, terjadi pengenalan wajah yang *false positive* seperti pada Gambar 16, dimana pengintip dikenali sebagai orang yang sama ditandai dengan label yang sama.



Gambar 16. Pengguna dan Pengintip Dikenali Sama

Faktor yang mempengaruhi terjadinya *false positive* ini ialah jumlah dan teknik pengambilan citra *dataset* yang masih belum terlalu tepat, sehingga jika kondisi wajah pengintip berubah secara seketika, terjadi kesalahan prediksi pengenalan wajah.

V. PENUTUP

A. *Kesimpulan*

Algoritma pengenalan wajah berhasil diimplementasikan dan berhasil mendeteksi ancaman *visual hacking* dalam pengujian berdasarkan desain eksperimen dan 5 skenario, dengan waktu kecepatan deteksi 2.7003 detik dan akurasi pengenalan wajah 94%.

B. *Saran*

Perlu adanya penelitian lanjutan untuk menentukan teknik atau metode terbaik, dalam menentukan citra *dataset* pengenalan pemilik komputer, penggunaan algoritma, maupun perangkat keras yang digunakan, dalam hal mendeteksi *visual hacking.*

DAFTAR PUSTAKA

[1] Ponemon, Larry dan tim. 2015. "*3M Visual Hacking Experiment*". Ponemon Institute *Research Repor*t.

[2] Ponemon, Larry dan tim. 2015. "*New Study Exposes Visual Hacking as Under-Addressed Low-Tech Threat*". Ponemon Instiute *whitepaper about visual hacking experiment*.

[3] Ponemon, Larry dan tim. 2015. "*New Study Exposes Visual Hacking Is A Global Problem*". Ponemon Institute *whitepaper about visual hacking experiment*.

[4] P. Viola, et al., "*Rapid Object Detection Using a Boosted Cascade of Simple Features*", *in Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. CVPR 2001, Kauai, Hawaii, 2001.

[5] T. Ojala, et al., "*Multiresolution gray-scaleand rotationinvariant texture classification with local binary patterns*", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, 2002.

[6] Matthew A. Turk, et al., "*Face Recognition Using Eigenfaces*", *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1991.

[7] Chi Ho Chan, "*Multi-scale Local Binary Pattern Histogram for Face Recognition*", *School of Electronics and Physical Sciences*, *University of Surrey*, Guildford, Surrey, 2008.

[8] Chakka Mounica, et al., "*FACE DETECTION AND RECOGNITION USING LBPH*", Vol. 5, No. 3, Aug., 2016.

[9] OpenCV, (2017) About [online]. Available : http://opencv.org/about.html.

SEKILAS TENTANG PENULIS

Luisan William Alexander, lahir di Manado pada tanggal 08 Maret 1995. Penulis menempuh pendidikan Sekolah Dasar di SD Garuda Manado dan kemudian melanjutkan pendidikan ke Sekolah Menengah Pertama di SMP N 2 Manado.Selanjutnya, penulis melanjutkan dan menyelesaikan pendidikan Sekolah Menengah Atas di SMA N 1 Manado dan pada saat penelitian ini dilakukan, penulis merupakan seorang mahasiswa Teknik Informatika Universitas Sam Ratulangi tahun 2012-2017. Selama masa perkuliahan penulis mengikuti kerja praktek di CROSSNET Indonesia cabang Manado dan pada masa akhir perkuliahan penulis bekerja di UPT-TIK UNSRAT sebagai tim pengembang.